

Remarks and Arguments

The specification has been objected to for containing embedded hyperlinks at page 8, lines 3 and 21. In response, the hyperlinks have been deleted.

Claims 5 and 13 were rejected under 35 U.S.C. §112, first paragraph as not being enabled by the disclosure. In particular, the examiner points to the storage volume interface layer recited in claims 5 and 13 as being mentioned in the specification at page 7, but not fully disclosed in a manner sufficient to enable the entire claim. The storage volume interface layer and the additional data service layers (network data replication, data imaging and data caching) are illustrated in Figure 1A and described in connection with Figure 1A on page 5, line 16 to page 6, line 20. As indicated there, the storage volume interface layer and the additional data service layers are kernel drivers that are implemented in platform-specific code that can necessarily be accessed from, and control, only the host on which they are installed. Consequently, as described at page 9, line 20 to page 10, line 5 a native interface layer is used to convert the platform-specific code into a platform independent language such as Java™. The platform independent language is in turn converted into an object-oriented model by the management facade. The specification further describes these layers in detail and their method of operation. For example, the interaction of the federated beans, the management facade and the native interface to set up and control a storage network data replication (SNDR) system is described in detail at page 13, line 29, to page 21, line 24, in connection with Figures 6-9.

Consequently, it is believed that the specification describes the storage volume interface layer and the additional data service layers and the manner of using them in sufficient detail to enable one skilled in the art to make and use the invention. Accordingly, the rejection of claims 5 and 13 under 35 U.S.C. §112, first paragraph is hereby traversed.

Claims 1, 3, 5-9, 11, 13-17 and 19-26 have been rejected under 35 U.S.C. §103(a) as obvious over an article entitled "Enterprise JavaBeans Technology Server Component Model for the Java Platform", Anne Thomas, pages 1-24 (Thomas) in view of U.S. Patent No. 6,442,541 (Clark.) The examiner asserts that Thomas teaches basic Java Beans operation including the manner of interacting with Java Beans and the

lookup service. However, the examiner admits that Thomas does not disclose controlling data passing between the driver software and the storage device to implement a data service. However, the examiner claims that the Clark patent discloses a database access mechanism in which Java Beans are used to control data passing between driver software and a storage device.

The present invention relates to the provision of management of data services by a three-tiered system. The lowest tier comprises management facade software running on each machine that converts the platform-dependent interface written with the low-level kernel routines to platform-independent method calls. These low level routines are located between the storage device driver and the storage device that the driver controls. The kernel routines intercept and control I/O information that passes between the driver and the device in order to provide the data services. However, because the kernel routines are at a very primitive and simple level, they must be combined and coordinated to provide the overall data service. Consequently, the middle tier is a set of federated Java Beans that communicate with the management facades to combine and coordinate the primitive services and also communicate with the upper tier of the system. The upper tier of the inventive system comprises presentation programs that can be directly manipulated by management personnel to view and control the system.

The Thomas article is a general discussion of the Java Bean framework and the overall operation and architecture of a Java Bean system. Accordingly, it illustrates the general architecture of the Java Bean system used to implement the present invention. However, the Thomas article does not discuss or suggest how to implement any specific system and, more particularly, does not teach or suggest how a data service system would be implemented with a general Java Beans architecture.

In order to remedy this difficulty the examiner suggests combining the teachings of the Thomas article with those of the Clark patent. The Clark patent discloses a system that uses Java Beans to retrieve information from a database. In particular, the Java Beans formulate a query to a database driver to retrieve information and then reformat the query information as necessary. Thus, according to the examiner, the Thomas and Clark combination would produce a system in which the enterprise Java

Bean architecture of Thomas could be used to control the database driver disclosed in Clark to provide database and other data services.

However, the Clark reference works by using Java Beans to control a database driver to provide the database services. The Java Beans do not intercept or control I/O information or data moving between the driver and the database. Controlling the driver as taught by Clark is effective in that application because the driver is designed to provide the desired database services. However, the driver software cannot be controlled to provide the desired services in the instant application because the driver software is not designed to provide these services. Thus, it is not possible to control a driver to provide the services, especially in those case where two storage devices are involved. Instead, the present system intercepts and manipulates the data passing between the driver software and its associated storage device. This arrangement allows the Java Beans architecture to directly control the storage device while still allowing the storage device to be accessed by other applications via the driver software as illustrated in Figures 1 and 2.

Independent claims 1, 9, 17 and 23 have been amended to more particularly point out this difference. For example, amended claim 1 recites the step of “running, in the host, management facade software that receives data from the driver software and the storage device and controls how that data passes between the driver software and the storage device and that retrieves attributes of the data service...” Neither the Thomas article nor the Clark patent teaches or suggests directly controlling the data flow between a driver and its associated storage device to provide data storage services, therefore the combination cannot teach or suggest this. Consequently, claims 1, 9, 17 and 23 patentably distinguish over the cited combination of references.

Claims 3 and 5-8 are dependent on, and incorporate the limitations of independent claim 1. Therefore, they patentable distinguish over the cited combination of references in the same manner as claim 1. In addition, these claims recite additional limitation not taught or suggested by the cited combination of references. For example, claim 5 recites “inserting a storage volume interface layer between the driver software and the storage device, inserting an additional data service interface layer between the storage volume interface layer and the storage device and using the storage volume

interface layer to divert information passing between the driver software and the storage device to the additional data service layer". As previously mentioned Thomas discloses only a general Java Beans architecture and Clark discloses controlling the driver rather than diverting the data passing between the driver and the storage device and passing the diverted data through another data services layer. Consequently, claim 5 also distinguishes over the cited combination of references.

Similarly, claim 7 recites using a lookup service to obtain a proxy to a bean in a first host and then using that proxy to retrieve another proxy of a bean in another host. Claim 8 recites that the first bean is used to control the second bean from the first host. This operation is used in the set up of a network data replication system. The Thomas reference discloses a lookup service to retrieve an object interface of a Java Bean but does not disclose or suggest that the object interface could be used to retrieve a second bean proxy or control the second bean. The examiner suggests that this operation could result from the relationship between data, but no data that might cause such an operation is disclosed in Thomas. Consequently, claims 7 and 8 patentably distinguish over the combination of Thomas and Clark.

Claims 11 and 13-16 are dependent on, and incorporate the limitations of independent claim 9. Therefore, they patentably distinguish over the cited combination of references in the same manner as claim 9. In addition, these claims recite additional limitations not taught or suggested by the cited combination of references. For example, claims 13 and 15-16 contain limitations that parallel those in claims 5 and 7-8, respectively. Consequently, they distinguish over the cited combination in the same manner as claims 5 and 7-8.

Claims 21-22 and 25-26 are dependent on, and incorporate the limitations of independent claim 17. Therefore, they patentably distinguish over the cited combination of references in the same manner as claim 17. In addition, these claims recite additional limitations not taught or suggested by the cited combination of references. For example, claims 21-22 and 25-26 contain limitations that parallel those in claims 7-8, respectively. Consequently, they distinguish over the cited combination in the same manner as claims 7-8.

Claims 2, 10 and 18 have been rejected under 35 U.S.C. §103(a) as obvious over Thomas in view of Clark and further in view of U.S. Patent No. 6,629,128 (Glass.) The examiner comments that the combination of Thomas and Clark teaches the recited limitations except that the combination does not teach using a command line interface to control the system. However the examiner asserts that the Glass patent shows such an interface and it would have been obvious to combine Glass with Thomas and Clark because the combination would permit the java beans to be controlled with a command line.

The Glass patent discloses a system for distributed processing in a computer network using distributed objects that uses code generators to automatically generate the required local and remote objects to perform the processing. Although it does mention command line interfaces, the reference is not concerned with providing data storage services via a Java Beans architecture. Consequently, its combination with Thomas and Clark cannot remedy the differences between the Thomas and Clark combination and the present invention. Claims 2, 10 and 18 are dependent on, and incorporate the limitations of independent claims 1, 9 and 17, respectively. Therefore, these claims patentable distinguish over the cited combination of references in the same manner as claims 1, 9 and 17.

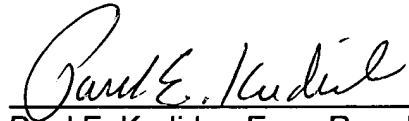
Claims 4 and 12 have been rejected under 35 U.S.C. §103(a) as obvious over Thomas in view of Clark and further in view of U.S. Patent No. 5,794,013 (McBrearty). The examiner comments that the combination of Thomas and Clark teaches the recited limitations except that the combination does not teach inserting a SCSI terminal emulation interface layer between the driver software and the storage device in order to make the storage device appear as a SCSI device. However the examiner asserts that the McBrearty patent discloses a SCSI emulator inserted between driver software and a remote storage. The examiner concludes it would have been obvious to combine McBrearty with Thomas and Clark because the combination would permit a storage device to operate in a SCSI environment.

The McBrearty patent discloses an emulator that makes one SCSI device (a hard drive) appear to be another SCSI device (a CD-ROM.) This emulation is used to allow software code on the hard drive to be tested without waiting for the code to be burned

into CD-ROMs. In the present invention the SCSI emulation layer is used to make non-SCSI devices appear to be SCSI devices. Claims 4 and 12 have been amended to recite that the storage device is a non-SCSI device and that the emulation layer is inserted in order to make the device appear to be a SCSI device. Such an emulation layer is not shown in Thomas, Clark or McBrearty. Consequently, claims 4 and 12 patentably distinguish over the combination of these references.

In light of the forgoing amendments and remarks, this application is now believed in condition for allowance and a notice of allowance is earnestly solicited. If the examiner has any further questions regarding this amendment, he is invited to call applicants' attorney at the number listed below. The examiner is hereby authorized to charge any fees or direct any payment under 37 C.F.R. §§1.17, 1.16 to Deposit Account number 02-3038.

Respectfully submitted



Date: 11/17/04

Paul E. Kudirka, Esq. Reg. No. 26,931
KUDIRKA & JOBSE, LLP
Customer Number 021127
Tel: (617) 367-4600 Fax: (617) 367-4656